

## # ADORDD

\*\*\*\*\*

¡ADORDD para (x) Harbour ya está disponible!

¡POR FAVOR HAGA UNA DONACIÓN!

Todos nos vamos a beneficiar seguro de este desarrollo, así que creo que es justo pedir a todos a contribuir con un mínimo de 20 Euros (debido a los costos de paypal) para una buena causa.

Visitar <http://ajusera.com/>, y allí desplácese hacia abajo y haga clic en el botón de PayPal "*Doar*". Se puede enviar un recibo de la contribución si se le envía un correo electrónico con sus datos.

Se trata de una organización sin fines de lucro que lleva a cabo una importante labor social el apoyo a la tercera edad, adolescentes problemáticos, la distribución de alimentos a los necesitados, etc. Esta organización se basa en la obra del Padre Jerónimo Usera (español) muy conocido en España, Portugal y América del Sur.

Puede obtener más información en [ajusera.com](http://ajusera.com) (lo siento, su único idioma es portugués).

Pasos para empezar a trabajar con ADORDD:

=====

**Paso 1:** Añada `adordd.prg` a su proyecto e incluya asimismo `adordd.ch`.

**Paso 2:** Establezca los parámetros como en `tryadordd.prg` (ver en `adordd.ch` la sintaxis) en su `main.prg`, añadiendo asimismo:

```
//  
// Cargamos ADORDD  
REQUEST ADORDD, ADOVERSION, RECSIZE  
  
RddRegister( "ADORDD", 1 )  
RddSetDefault( "ADORDD" )
```

### A) SET ADO TABLES INDEX LIST TO...

-----

Este comando es utilizado a nivel SQL para construir SELECT con la cláusula order by.

Así, los campos deben estar separados por comas y se pueden incluir funciones SQL o las cláusulas DESC u ASC. Este comando NO puede incluir funciones de Clipper/(x)Harbour porque son desconocidas para SQL.

Ejemplo:

```

SET      ADO      TABLES      INDEX      LIST      TO      {
{"TABLE1", {"FIRST", "FIRST      DESC"}      },      {"TABLE2"
, {"CODID", "CODID"} } }

```

## B) SET ADODBF TABLES INDEX LIST TO...

---

Este comando es utilizado por ADORDD para evaluar expresiones de Clipper/(x)Harbour tales como:

```
&( indexkey( 0 ) )
```

```
OrdKey( )
```

Etc

Por lo tanto, debe contener sus expresiones de índice reales.

Ejemplo:

```

SET      ADODBF      TABLES      INDEX      LIST      TO      {
{"TABLE1", {"FIRST", "FIRST"}      },      {"TABLE2"
, {"CODID", "STR(CODID, 2, 0)"} } }

```

## C) SET ADO TEMPORAY NAMES INDEX LIST TO...

---

Este comando indica los nombres utilizados para los archivos temporales a nivel de SQL.

Debe empezar por TMP o TEMP, pero puede ser "TMPROGER"

Estos archivos temporales se utilizan principalmente para los índices temporales creados en el server SQL como TEMPORALES y automáticamente son destruidos después de que termine la conexión.

Únicamente son visibles para el usuario que los creó.

Ejemplo:

```
SET ADO TEMPORAY NAMES INDEX LIST TO {"TMP", "TEMP"}
```

## D) SET ADO FIELDRECNO TABLES LIST TO...

---

Este comando le permite indicar un campo autoincremental diferente para *cada* tabla que es utilizado como base para Recno()

Ejemplo:

```
SET      ADO      FIELDRECNO      TABLES      LIST      TO
{ { "TABLE1 " , "HBRECNO " } , { "TABLE2 " , "HBRECNO " } }
```

#### **E) SET ADO DEFAULT RECNO FIELD TO ...**

-----

Este comando indica el nombre de campo predeterminado autoincremental que se utilizará como RECNO en *todas* las tablas.

Ejemplo:

```
SET ADO DEFAULT RECNO FIELD TO "HBRECNO"
```

#### **ATENCIÓN:**

Los comandos D y E son absolutamente necesarios y sin ellos la navegación con ADORDD pueden ser impredecible.

#### **F) SET ADO DEFAULT DATABASE TO... SERVER TO ... ENGINE TO ACCESS/MYSQL/MSSQL USER TO ... PASSWORD TO ...**

-----

Este comando especifica el servidor predeterminado y base de datos y parámetros de autenticación que estamos utilizando.

La conexión se establece aquí.

Los motores soportados en ENGINE son ACCESS, MYSQL, ORACLE, INFORMIX, MSSQL, FIREBIRD, POSTGRE, ANYWHERE, DBASE, SQLITE, FOXPRO y, en fin, ADS.

Ejemplo:

```
// Access
SET ADO DEFAULT DATABASE TO "D:\LUCAS\TEST2.mdb"
SERVER TO " " ENGINE TO ACCESS USER TO " " PASSWORD TO
" "
```

```
// MySQL
SET ADO DEFAULT DATABASE TO cDataBase SERVER TO
cServer ENGINE TO "MYSQL" USER TO cUser PASSWORD TO
cPassWord
```

#### **G) SET ADO LOCK CONTROL SHAREPATH TO...**

-----

Este comando permite ADORDD asegurar los registros de bloqueo y el uso exclusivo de los archivos como cualquier otro RDD.

**Por defecto, no es necesario activarlo, ya que se encarga el motor SQL de la gestión de los bloqueos. Por ello, debemos poner:**

```
SET ADO FORCE LOCK OFF
```

Debe suministrar una ruta donde ADORDD crea el archivo tlocks.dbf para controlar esta funcionalidad.

Este archivo RDD debe ser un RDD capaz de trabajar con bloqueos como es el caso de DBFCDX.

Esto no es una tabla de SQL, y si usted necesita para trabajar en la WAN y la necesidad de usar esta característica tendrá que disponer de a), la conexión con el servidor SQL y b), establecer una VPN donde se puede acceder a este recurso compartido.

Ejemplo:

```
SET ADO LOCK CONTROL SHAREPATH TO "D:\LUCAS" RDD TO  
"DBFCDX"
```

**Paso 3:** Ya puede trabajar con ADORDD como con DBFCDX.

No debe hacer cambios de código en sus aplicaciones con la excepción de:

Todas las expresiones con variables para, por ejemplo, un índice las variables deben ser evaluadas antes de enviarlo a ADORDD.

Los registros borrados son inmediatamente borrados y no se pueden recuperar de nuevo. Por lo tanto, cualquier operación en los registros eliminados deben tener lugar antes de eliminar el registro o un error se producirá.

Es decir, operaciones tales como:

```
delete all
```

```
while...
```

```
if lconditon
```

```
recall record
```

Deben invertirse:

```
while....
```

```
if !lcondtion  
    delete record
```

Nada más.

Revisión del manual en Español: 1.0; Lucas de Beltrán; 16.06.2015